DIGITAL METHODS OF PRODUCING RANDOM AND VARIABLE TIME INTERVALS¹

NORMAN T. WELFORD

FELS RESEARCH INSTITUTE

Accurate time intervals which can be varied at will are often required in psychological experiments. When it is desired to vary the time intervals frequently, randomly, or rapidly one after the other, the usual analog devices such as electromechanical or electronic timers, film-strip program-timers (Scott, 1965) and audio type magnetic tape programmers (Buckhout, 1965; Kintz, 1965) are often insufficiently accurate or flexible.

Systems based on a digital interval timer are relatively expensive but extremely versatile, highly accurate and reliable. Solid state logic elements combined with a punched paper tape reader provide a most flexible way to implement this function. These systems also can be programmed from tapes generated by a computer. This article describes in detail one especially convenient method and outlines advantages and disadvantages of other methods.

The basic system uses a counting method and is composed of a source of clock-pulses, some control logic and a paper tape reader. The time intervals desired are punched as numbers on the paper tape. For each time interval, the clock-pulses are gated by the control logic into the counter. When the number of clock-pulses counted is equal to the number punched on the tape, further control logic signals the end of the time interval. Each number on the tape represents the number of time intervals between clock-pulses required to total to the desired time period.

This basic system has been used as part of a programmer for presenting stimuli in a continuous choice reaction time task with human

subjects and in a general purpose stimulus presentation apparatus for work with infants. It would also be suitable for studies of the effect of a schedule of variable delays between a response and a discrete reinforcement such as brain stimulation.

The reliability depends entirely on the component parts of the system; a tape reader may be expected to make less than one error in a million readings and digital logic elements have thousands of hours between failures.

Other uses of solid state logic elements in programming systems for behavioral research have been described by Herrick and Denelsbeck (1963) and by Weiner (1963).

The Recommended Method Using Nine's Complement

Numbers representing the desired time intervals are punched in binary-coded decimal (BCD) form on paper tape. As the tape is read each number is converted into its nine's complement² by the use of logic elements. In this complement form the number is set into a count-up counter, consisting of a chain of flip-flops. At the start of a time interval, clock-pulses from the timing source enter the counter and add to the count already set in. The end of the time interval is signaled when the counter is reset to zero; this occurs with the receipt of the next pulse after the counter has reached its maximum count.

The use of the complement of the number and a count-up counter simplifies the logic hardware in the system. (See Alternate Methods below).

¹This research was supported by Grant M-623 from the National Institute of Mental Health, United States Public Health Service, Bethesda, Maryland. Reprints may be obtained from the author, Medical Electronics Laboratory, University of Texas Medical Branch, Galveston, Texas 77550.

²The nine's complement of a number is the result of subtracting a number from nine or a sequence of nines, e.g., the nine's complement of 235 = 999-235 = 764. The normal complement is the ten's complement which is the result of subtracting a number from 10, or a multiple of 10, e.g., the ten's complement of 235 = 1000-235 = 765.

The nine's complement, rather than the ten's complement, is used as this can be derived by simple logic from the number on the tape. Carrying is not required, so the same set of logic elements can be used to convert each digit of the number into its nine's complement form.

Because the nine's complement is used in this method, the time interval will be one count more than the number on the tape. The count is easily corrected by adding one pulse to the counter at the beginning of the time interval, or the number punched on the tape can be made one less than the time interval count desired. (When using certain time pulse circuits, such as Digibit MV 107, *etc.*, a pulse is emitted as soon as the unit is turned on, thus providing the extra pulse automatically.)

Figure 1 shows the decimal, binary and logic forms of the numbers 0-9 with their nine's complements.

The logic equations for the derivation of each binary digit of the nine's complement from the binaries of normal BCD numbers are:

Nine's complement binary 1 = $\bar{1}$ in normal BCD.

Nine's complement binary 2 = 2 in normal BCD.

Nine's complement binary 4 = $2 \cdot \bar{4} \vee \bar{2} \cdot 4$ in normal BCD.

Nine's complement binary 8 = $\bar{2} \cdot \bar{4} \cdot \bar{8}$ in normal BCD.

These equations can be understood by examining the table in Fig. 1 which shows the logical equivalents of the numbers 0-9 as normal BCD numbers and as their nine's complements. Every time 1 is present in the nine's complement form of a number, and only at this time, $\bar{1}$ is present in the normal form of this same number, so that 1 in the nine's complement is equivalent to $\bar{1}$ in the normal number. Every time there is 2 in the nine's complement there is 2 in the normal number. When, and only when, there is 4 in the nine's complement, there is either 2 and $\bar{4}$, or $\bar{2}$ and 4, in the normal number. When, and only when, there is 8 in the nine's complement, there is $\bar{2}$, $\bar{4}$, and $\bar{8}$ in the normal number.

The method of implementing the logical conversion of a number into its nine's complement is shown in Fig. 2.

It uses two two-leg AND gates, one two-leg OR gate (these form an exclusive OR gate) and one three-leg AND gate. As both the presence and absence of each hole in the tape requires a true logic level, four inverters will be needed to give $\bar{1}$, $\bar{2}$, $\bar{4}$, and $\bar{8}$ if these outputs are not available from the tape reader.

Numbers 0-9

Nine's Complements

Decimal Digit	Binary Code 8 4 2 1	Logical Equivalent	Decimal Digit	Binary Code 8 4 2 1	Logical Equivalent
0	0 0 0 0	$\bar{1} \cdot \bar{2} \cdot \bar{4} \cdot \bar{8}$	9	1 0 0 1	$1 \cdot \bar{2} \cdot \bar{4} \cdot \bar{8}$
1	0 0 0 1	$1 \cdot \bar{2} \cdot \bar{4} \cdot \bar{8}$	8	1 0 0 0	$\bar{1} \cdot \bar{2} \cdot \bar{4} \cdot \bar{8}$
2	0 0 1 0	$\bar{1} \cdot 2 \cdot \bar{4} \cdot \bar{8}$	7	0 1 1 1	$1 \cdot 2 \cdot 4 \cdot \bar{8}$
3	0 0 1 1	$1 \cdot 2 \cdot \bar{4} \cdot \bar{8}$	6	0 1 1 0	$\bar{1} \cdot 2 \cdot 4 \cdot \bar{8}$
4	0 1 0 0	$\bar{1} \cdot \bar{2} \cdot 4 \cdot \bar{8}$	5	0 1 0 1	$1 \cdot \bar{2} \cdot 4 \cdot \bar{8}$
5	0 1 0 1	$1 \cdot \bar{2} \cdot 4 \cdot \bar{8}$	4	0 1 0 0	$\bar{1} \cdot \bar{2} \cdot 4 \cdot \bar{8}$
6	0 1 1 0	$\bar{1} \cdot 2 \cdot 4 \cdot \bar{8}$	3	0 0 1 1	$1 \cdot 2 \cdot \bar{4} \cdot \bar{8}$
7	0 1 1 1	$1 \cdot 2 \cdot 4 \cdot \bar{8}$	2	0 0 1 0	$\bar{1} \cdot 2 \cdot \bar{4} \cdot \bar{8}$
8	1 0 0 0	$\bar{1} \cdot \bar{2} \cdot \bar{4} \cdot 8$	1	0 0 0 1	$1 \cdot \bar{2} \cdot \bar{4} \cdot \bar{8}$
9	1 0 0 1	$1 \cdot \bar{2} \cdot \bar{4} \cdot 8$	0	0 0 0 0	$\bar{1} \cdot \bar{2} \cdot \bar{4} \cdot \bar{8}$

Fig. 1. The numbers 0-9 and their nine's complements in Decimal, Binary and Logical Equivalent forms.

Fig. 2. Logic diagram of Nine's Complement Converter.

Details of the Elements of the System

Clock-pulses. These are generated by an oscillator which can be made to give pulses from many thousands per second down to one or fewer per minute. Simple multivibrator circuits have an accuracy of better than 5%, a unijunction oscillator (G. E. Transistor Manual, 1964) using high stability components better than 1%. Much higher accuracy and stability can be obtained from oscillators based on tuning forks.³

Counter. Each decade of the counter consists of four flip-flops, connected in a binary counting chain, with feedback so that every tenth pulse causes all four flip-flops to be reset to zero. This constitutes a binary-coded decade, and several decades can be cascaded to make a counter as large as required. It is arranged as a count-up or adding counter, that is, each pulse fed-in increases the count by one.

Paper tape reader. This may be photoelectric and easily read tape at 300 rows per sec, or mechanical and be limited to about 50 rows per sec. Each digit of the desired time interval is punched in binary-coded decimal form and occupies one row on the tape.

Setting a number into a counter. The number on the tape is set into the counter digit by digit. The logic levels from the tape reader which represent the presence or absence of the holes representing 1, 2, 4, and 8 for the digit being read, are connected to the nine's complement converter. The true logic levels from

this, which represent the conversion of the digit on the tape, are connected via AND gates to set the relevant flip-flops in the counter (one two-leg AND needed for each flip-flop).

A block diagram, Fig. 3, shows the interrelations of the component parts of the system discussed above. As each successive row on the tape representing a BCD number is read it is converted into its nine's complement and set into the counter. A pulse from the tape reader for each row also goes to the control logic and the appropriate AND gates are opened so that the hundreds, tens, and units are correctly set into the counter. At the start of the time interval the clock-pulses from the oscillator add to the count in the counter. When the counter has been filled and resets to zero, a pulse from the counter signals the end of the time interval.

Performance Capabilities

The three-digit system discussed above can produce time intervals over a range of 1000:1 in length. The minimum length of a time interval following a previous one without pause, will be dependent on the time taken for the tape reader to advance and set the new number in the counter. Four rows on the tape will be required, three for the time interval and one for a special character on the tape to separate each time interval. If a 300 rows-per-sec paper

Fig. 3. Block Diagram for Three Digit Timer.

³Time and Frequency, Batavia, Illinois, supply tuning forks at reasonable cost.

tape reader is used, then it will take about 5 msec to start the tape moving and 3.3 msec to continue moving the tape, while reading each of the three digits and the special character denoting the end of the number. This means a minimum time between intervals of 18.2 msec. If the clock-pulses are slower than 50 per sec, there will be sufficient time to advance the tape and set the numbers representing the next time interval into the counter before the next clock-pulse.

Alternate Methods

Punching complements on the tape. A count-up counter is used, as in the method discussed, but instead of deriving the nine's complement of the number punched on the tape, the tens' complement is punched on the tape. This number is directly set into the counter and the interval timed as before. This method has the advantage of simplicity of logic; however, the burden of having to punch the numbers on the tape in complement form is placed on the experimenter.

Using a subtracting counter. The counter is made to count down instead of up. The number on the tape is set in, and the clock-pulses now subtract until the count has been reduced to zero, when the proper time interval will have elapsed. This apparently attractive system has some disadvantages. First, when a flip-flop of a count-down counter is set, a carry is produced so that when setting-in the number from the tape, additional logic is necessary to prevent the carry pulses affecting the counter. Second, when the counter counts down to zero, the most significant flip-flop of the counting chain does not produce a carry pulse, so that the zero has to be recognized by an AND gate with four legs for each binary-coded decimal decade of the counter. (When a count-up counter reaches zero there is a carry pulse from the most significant flip-flop.)

Using a comparison technique. The count is increased during the time interval until it equals the number punched on the tape, at

which point logic circuits detect the equality and terminate the time interval. For this method the reading from the tape reader must be present throughout the time interval, so it can only be used for numbers represented by a single row of holes across the tape, unless use is made of a block tape reader which reads several rows at once, or the number is stored on a set of flip-flops. However, if more than a single row is used for the number, the equality detecting circuitry becomes cumbersome. (An exclusive OR gate consisting of two two-leg AND gates each feeding into a two-leg OR gate is required for each binary digit. The output of each exclusive OR gate then must be combined by a further AND gate, with four legs for each decimal digit.)

Stepping the tape reader at a fixed rate. This extremely simple method uses one hole on the tape to signal the start of the time interval and another hole to signal the end of the time interval. The length of the time interval depends on the length of the tape between the start and stop signals. If a wide range of time intervals is desired then the amount of tape required will be large. This can be handled if a tape spooler and rewinder unit is used with the tape reader. However, the preparation of such tapes may be tedious and have to be done by hand.

REFERENCES

- Buckhout, R. An inexpensive laboratory stimulus programmer. *J. exp. Anal. Behav.*, 1965, 8, 320.
- G. E. Transistor Manual, 7th Edition, 1964. Chapter 13. General Electric Co., Syracuse, N. Y.
- Herrick, R. M. and Denelsbeck, J. S. A system for programming experiments and for recording and analyzing data automatically. *J. exp. Anal. Behav.*, 1963, 6, 631-635.
- Kintz, R. T. The tape recorder as a programmer. *J. exp. Anal. Behav.*, 1965, 8, 294.
- Scott, K. G. A device for the timed punching of tapes for programmers. *J. exp. Anal. Behav.*, 1965, 8, 105-106.
- Weiner, H. Operant programming with transistorized digital elements. *J. exp. Anal. Behav.*, 1963, 6, 193-195.